

13015 计算机系统原理

第四章 课后习题 计算题讲解(部分)

讲师/公众号：**小飞学长Pro**

13015 计算机系统原理【第四章课后题】

3. 假设一个 C 语言程序有两个源文件：main.c 和 test.c，它们的内容如图 4.27 所示。

<pre>1 /* main.c */ 2 int sum(void); 3 int a[4]={1, 2, 3, 4}; 4 extern int val; 5 int main(){ 6 val=sum(), 7 return val; 8 }</pre>	<pre>1 /* test.c */ 2 extern int a[]; 3 int *ptr, val=0; 4 int sum(){ 5 int i; 6 for (i=0; i<4; i++) 8 val += a[i]; 9 return val; 10 }</pre>
--	---

图 4.27 题 3 图

对于编译生成的可重定位目标文件 test.o，填写表 4.1 中各符号的情况，说明每个符号是否出现在 test.o 的符号表 (.symtab 节) 中，如果是的话，则定义该符号的模块是 main.o 还是 test.o、该符号的类型相对于 test 模块是全局、外部还是本地符号、该符号出现在 test.o 中的哪个节 (.text、.data 或 .bss) 或哪个特殊伪节 (ABS、UNDEF 或 COMMON) 中。

13015 计算机系统原理【第四章课后题】

3. 假设一个 C 语言程序有两个源文件：main.c 和 test.c，它们的内容如图 4.27 所示。

```
1  /* main.c */
2  int sum(void);
3  int a[4]={1, 2, 3, 4};
4  extern int val;
5  int main(){
6      val=sum(),
7      return val;
8  }
```

```
1  /* test.c */
2  extern int a[];
3  int *ptr, val=0;
4  int sum(){
5      int i;
6      for (i=0; i<4; i++)
7          val += a[i];
8      return val;
9  }
```

符号	在test.o的符号表中?	定义模块	符号类型	节
a	在	main.o	外部	UNDEF
ptr	在	test.o	全局	COMMON
val	在	test.o	全局	.bss
sum	在	test.o	全局	.text
i	不在	-	-	-

13015 计算机系统原理【第四章课后题】

4. 假设一个 C 语言程序有两个源文件：main.c 和 swap.c，其中，main.c 的内容如图 4.9a 所示，而 swap.c 的内容如下：

```
1 extern int buf[ ];
2 int *bufp0 = &buf[ 0 ];
3 int *bufp1;
4
5 static void incr() {
6     static int count=0;
7     count++;
8 }
9 void swap() {
10    int temp;
11    incr();
12    bufp1 = &bufp[ 1 ];
13    temp = *bufp0;
14    *bufp0 = *bufp1;
15    *bufp1 = temp;
16 }
```

```
1 extern void swap(void);
2
3 int buf[2] = {1, 2};
4
5 int main(){
6     swap();
7     return 0;
8 }
```

a) main.c 文件

13015 计算机系统原理【第四章课后题】

对于编译生成的可重定位目标文件 swap.o，填写表 4.2 中各符号的情况，说明每个符号是否出现在 swap.o 的符号表 (.symtab 节) 中，如果是，定义该符号的模块是 main.o 还是 swap.o、该符号的类型相对于 swap.o 是全局、外部还是本地符号、该符号出现在 swap.o 中的哪个节 (.text、.data 或 .bss) 或哪个特殊伪节 (ABS、UNDEF 或 COMMON) 中。

符号	在 <u>swap.o</u> 的符号表中?	定义模块	符号类型	节
buf	在	main.o	外部	UNDEF
bufp0	在	swap.o	全局	.data
bufp1	在	swap.o	全局	COMMON
incr	在	swap.o	本地	.text
count	在	swap.o	本地	.bss
swap	在	swap.o	全局	.text
temp	不在	-	-	-

13015 计算机系统原理【第四章课后题】

6. 以下每一小题给出了两个源程序文件，它们被分别编译生成可重定位目标模块 $m1.o$ 和 $m2.o$ 。在模块 m_j 中对符号 x 的任意引用与模块 m_i 中定义的符号 x 关联记为 $REF(m_j.x) \rightarrow DEF(m_i.x)$ 。请在下列空格处填写模块名和符号名，以说明给出的引用符号所关联的定义符号，若发生链接错误则说明其原因；若从多个定义符号中任选则给出全部可能的定义符号，若是局部变量则说明不存在关联。

(1)

```
/* m1.c */
int p1(void);
int main()
{
    int p1 = p1();
    return p1;
}

/* m2.c */
static int main=1;
int p1()
{
    main++;
    return main;
}
```

答:

- ① $REF(m1.main) \rightarrow DEF(\underline{\hspace{2cm}}.\underline{\hspace{2cm}})$
- ② $REF(m2.main) \rightarrow DEF(\underline{\hspace{2cm}}.\underline{\hspace{2cm}})$
- ③ $REF(m1.p1) \rightarrow DEF(\underline{\hspace{2cm}}.\underline{\hspace{2cm}})$
- ④ $REF(m2.p1) \rightarrow DEF(\underline{\hspace{2cm}}.\underline{\hspace{2cm}})$

- ① $REF(m1.main) \rightarrow$ 在m1中不存在对main的引用
- ② $REF(m2.main) \rightarrow DEF(m2.main)$
- ③ $REF(m1.p1) \rightarrow DEF(m2.p1)$
- ④ $REF(m2.p1) \rightarrow$ 在m2中不存在对p1的引用

13015 计算机系统原理【第四章课后题】

(2)

```
/* m1.c */
int x=100;
int p1(void);
int main()
{
    x=p1();
    return x;
}

/* m2.c */
float x=100.0;
int main=1;
int p1()
{
    main++;
    return main;
}
```

- ① REF(m1. main)→DEF(_____ . _____)
- ② REF(m2. main)→DEF(_____ . _____)
- ③ REF(m1. x)→DEF(_____ . _____)

答：

发生链接错误，因为全局变量 **x** 在两个目标模块中都有强定义

13015 计算机系统原理【第四章课后题】

(3)

```
/* m1.c */          /* m2.c */
int p1(void);       int x=10;
int p1;             int main;
int p1()            int main()
|
|
|                   |
int x=p1();         main=1;
return x;           return x;
|                   |
|                   |
```

答:

- ① REF(m1.main)→DEF(_____ . _____)
 - ② REF(m2.main)→DEF(_____ . _____)
 - ③ REF(m1.p1)→DEF(_____ . _____)
 - ④ REF(m1.x)→DEF(_____ . _____)
 - ⑤ REF(m2.x)→DEF(_____ . _____)
- ① REF(m1.main)→在m1中不存在对main的引用
 - ② REF(m2.main)→DEF(m2.main)
 - ③ REF(m1.p1)→
 - ④ REF(m1.x)→ 局部变量, 不存在关联
 - ⑤ REF(m2.x)→DEF(m2.x)

13015 计算机系统原理 【第四章课后题】

(4)

```
/* m1.c */
int p1(void);
int x, y;
int main()
{
    x=p1();
    return x;
}

/* m2.c */
double x=10;
int y;
int p1()
{
    y=1;
    return y;
}
```

- ① REF(m1.x)→DEF(____.____)
- ② REF(m2.x)→DEF(____.____)
- ③ REF(m1.y)→DEF(____.____)
- ④ REF(m2.y)→DEF(____.____)

答:

- ① REF(m1.x)→DEF(m2.x)
- ② REF(m2.x)→在m2中不存在对x的引用
- ③ REF(m1.y)→在m1中不存在对y的引用
- ④ REF(m2.y)→DEF(m1.y) 或者DEF(m2.y)

谢谢大家